REMARKS

This application has been carefully considered in connection with the Examiner's Action. Reconsideration and allowance are respectfully requested in view of the following.

While, as Original, Claims 1, 8 and 17 were patentably distinguishable over the cited art, by this Amendment, the Applicant has amended each of these claims such that the distinguishing features are more clearly recited. Claims 2, 4-5, 11-13, 18 and 21 have been amended to be more consistent with amended Claims 1, 8 or 17. Finally, new Claims 22-32, all of which are directed to still further patentably distinguishable embodiments of the invention, have been added.

Claims 1-3, 8-10 and 17 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Ronen *et al.*, U.S. Patent No. 6,675,376 (hereafter Ronen) and Claims 4-7, 11-16 and 18-21 stand rejected under 35 U.S.C. § 103(a) as unpatentable over Ronen in view of Blaner *et al.*, U.S. Patent No. 6,029,240 (hereafter Blaner). In response, the Applicant respectfully traverses the various rejections of Claims 1-21 and instead submits that Claims 1-21, as above amended, as well as newly added Claims 22-29, are neither taught nor suggested by the art of record. Accordingly, the reconsideration and withdrawal of the various rejections of Claims 1-21 and the allowance of Claims 1-29, as above amended, are respectfully requested.

As above amended, Claim 1 is directed to a method for operating a pipeline in a processor. In accordance with the claimed method, a number of unfused instructions are fetched. A plurality of the fetched unfused instructions are then combined into a

control word which can be processed by one execution unit. Importantly, the number of unfused instructions that are fetched is greater than the number of execution units in the processor. Similarly, Claim 8 is directed to a processor having an instruction pipeline which includes first and second stages. The first stage fetches and decodes a number of unfused instructions. Fusing logic in the second stage combines a plurality of the decoded unfused instructions into a control word capable of being executed by one execution unit. Again, importantly, the number of unfused instructions fetched and decoded by the first stage exceeds the number of execution units in the processor. Finally, Claim 17 is directed to a system for coupling instructions from memory to execution units in a processor. The system includes fetching means for fetching and decoding a number of unfused instructions and fusing means for combining a plurality of the unfused instructions into a control word which can be executed by one execution unit. As before, the number of unfused instructions fetched and decoded by the fetching means exceeds the number of execution units in the processor.

The prior art has been carefully considered but neither teaches nor suggests Applicant's invention as defined in Claims 1-32, as above amended. The Examiner cites Ronen as substantially teaching the claimed invention, including a data processing system comprising: a) pipeline processor, b) means and method for fetching a number of instructions, and c) means and method for combining a plurality of instructions into a control word. While acknowledging that Ronen does not expressly detail that the number of instructions fetched may be greater than the number of execution units, the Examiner asserts that, because Ronen teaches three execution units (floating point execution unit, integer execution unit and fused execution unit) and because the fused

instruction unit is capable of executing a fused instruction comprised of the combination of two fetched instructions, the system disclosed by Ronen includes the "capability of executing a total of four instructions using three execution units." The Examiner further asserts that "[o]ne of ordinary skill therefore would have been motivated of fetching four instructions so that the system would have executed the instructions at maximum efficiency and maximum capacity. Fetching four instructions would have comprised more instructions than the number of execution units that comprised three execution units."

The Applicant respectfully disagrees with the Examiner's characterization of Ronen and instead submits that, upon properly characterizing Ronen, Applicant's invention, as defined by Claims 1-32, as above amended, is neither be taught nor suggested by Ronen, either alone or in combination with Blaner. Accordingly, the Applicant respectfully requests the reconsideration and withdrawal of the rejection of Claims 1-21 and the allowance of Claims 1-32, as above amended.

While the Examiner properly notes that Ronen teaches certain aspects of the fusion of multiple instructions into a single, fused, instruction, by asserting that Ronen teaches fetching four instructions for execution by three execution units, the Examiner is erroneously suggesting that the fusing of instructions occurs after fetching. However, a careful reading of Ronen clearly reveals otherwise. More specifically, Ronen specifically teaches that fusion takes place in the main memory, before the instructions are fetched for execution. Thus, using the Examiner's logic, at best, Ronen may be characterized as disclosing the fetching of three instructions for execution by three execution units and may not be characterized as teaching or suggesting a method,

processor or system in which the number of instructions fetched exceeds the number of execution units.

In support of Applicant's characterization of Ronen, the Examiner's attention is directed to col. 5, lines 47-60 of Ronen, which states that:

> "An optimizing compiler or a user using, for example, a code development tool, may produce the fused instructions. A fused instruction generator of the optimizing compiler analyzes the instructions to find simple, dependent instructions. If two simple, dependent instructions are found, then those two instructions are fused to create one fused instruction and the two simple, dependent instructions are replaced in the program with the fused instruction. The fused instruction generation continues performing this analysis until all simple, dependent instructions are fused together. In addition, the user may manually analyze the program instructions (using, for example, a code development tool) and upon encountering two simple, dependent instructions, replaces those two instructions with the fused instruction (emphasis added by Applicant)."

In further support of the foregoing interpretation of Ronen, the Examiner's attention is directed to col. 3, lines 54-56, which states that "the number of instructions in the program [is reduced] by fusing simple dependent instructions" and to col. 7, lines 55-57 which states that "[w]hen the decode unit decodes a fused instruction, the fused instruction is separated into one or two different uops." See also col. 8, lines 15-17.

Thus, any teachings by Ronen of the fusing of instructions is clearly directed to techniques for reducing the size and improving the efficiency of a program stored in memory and not, as contemplated by the Applicants, to a method and apparatus for processing instructions which includes the fetching and decoding of a number of unfused instruction which exceeds the number of execution units and combining a plurality of the fetched unfused instructions into a control word. By doing so, the Applicant enables the processor to fetch more instructions than it is capable of

executing simultaneously and subsequently combine selected ones of the fetched instructions so that processor can simultaneously execute the reduced number of instructions.

By this Amendment, the Applicant has amended independent Claims 1, 8 and 17 to more clearly recite this distinction. More specifically, the processing method defined by Claim 1 has been amended to clearly state that the fetched instructions are unfused and that the number of fetched unfused instructions is greater than the number of execution units in the processor. Similarly, the processor defined by Claim 8 has been amended to clearly state that the first stage fetches and decodes a number of unfused instructions and that the number of unfused instructions fetched and decoded by the first stage is greater than the number of execution units Finally, the system defined by Claim 17 has been amended to clearly state that that the fetching means fetch and decode a number of unfused instructions, that the number of unfused instructions fetched and decoded by the fetching means is greater than the number of execution units in the processor and that the fusing means combines a plurality of the fetched and decoded unfused instructions into a control unit which can be executed by one execution unit.

Claims 2-7 and 22-25 are dependent on independent Claim 1. Accordingly, the Applicant respectfully submits that Claims 2-7 and 22-25 are patentably distinguishable over Ronen and/or the proposed combination of Ronen and Blaner for the reasons set forth above with respect to Claim 1. Claims 9-16 and 26-28 are all dependent on independent Claim 8. Accordingly, the Applicant respectfully submits that Claims 9-16 and 26-28 are patentably distinguishable over Ronen and/or the proposed combination

of Ronen and Blaner for the reasons set forth above with respect to Claim 8. Claims 18-21 and 29-32 are all dependent on independent Claim 17. Accordingly, the Applicant respectfully submits that Claims 18-21 and 29-32 are patentably distinguishable over Ronen and/or the proposed combination of Ronen and Blaner for the reasons set forth above with respect to Claim 17.

Further by this Amendment, the Applicant has added new Claims 22-32, all of which, as previously noted, are directed to still further embodiments of the invention patentably distinguishable over the cited art. It is further noted that a number of the newly added claims recite additional features of Applicant's claimed method, processor and system neither taught nor suggested by Ronen. By way of example, Claims 22, 26 and 29 further recite that, when combining a plurality of unfused instructions into a control word, the unfused instructions are to be non-conflicting instructions capable of being executed simultaneously. Further by way of Example, Claims 24, 28 and 31 further recite that, when combining a plurality of unfused instructions into a control word, the unfused instructions are instructions which modify a common register. In contrast, Ronen discloses only the fusing of "simple, dependent instructions", i.e., where the destination operand of a first instruction is a source operand of a second instruction. Nowhere does Ronen contemplate the fusing of instructions such as those defined in new Claims 22-32. Accordingly, the Applicant further submit that new Claims 22-32 are further distinguishable over Ronen.

For all of the above reasons, the Applicant respectfully requests the reconsideration and withdrawal of the various rejections of Claims 1-21 and the allowance of Claims 1-32, as above amended.

This application is now considered to be in condition for allowance. A prompt Notice to that effect is, therefore, earnestly solicited.

Respectfully submitted,

CONLEY ROSE, P.C.

Date: August 3, 2004

5700 Granite Parkway, Suite 330
Plano, Texas 75024
Telephone: (972) 731-2288
Facsimile: (972) 731-2289

Michael S. Bush
Reg. No. 31,745

ATTORNEY FOR APPLICANT